REMARKS/ARGUMENTS

Claims 1-5, 7-10 and 12-14 are pending in the present application. Claims 1, 3-4, 7, 12, and 14 were amended; and claims 15-16 were added. Reconsideration of the claims is respectfully requested.

I. 35 U.S.C. § 103, Obviousness

I.A. Claims 1, 4-5, 12 and 14 over *O'Grady* in view of *Jaquette* and *Fish*

The Examiner has rejected claims 1, 4-5, 12 and 14 under 35 U.S.C. § 103(a) as being unpatentable over *O'Grady* et al., Method and Apparatus for Reordering Packet Data Units in Storage Queues for Reading and Writing Memory, U.S. Patent No. 6,757,791, dated June 29, 2004 (hereinafter referred to as "*O'Grady*") in view of *Jaquette* et al., ECC in Memory Arrays Having Subsequent Insertion of Content, U.S. Patent No. 6,009,547, dated December 28, 1999 (hereinafter referred to as "*Jaquette*") and *Fish* et al., Disk Data Control, U.S. Patent No. 4,241,420, dated December 23, 1980 (hereinafter referred to as "*Fish*"). This rejection, as it might be applied to the claims as amended, is respectfully traversed.

Applicants have amended claims 1 and 12 to describe connecting a protocol interface device between a first device that uses the first protocol and a second device that uses the second protocol. The protocol interface device includes a cyclical redundancy code (CRC) engine. The Examiner relies on *Jaquette*, CRC generator circuit 20, to teach a CRC engine.

Applicants have also amended claims 1 and 12 to describe if the CRC engine determines that an end one of said plurality of blocks that includes an end of said transfer length is not full of data, adding padding. Regarding the previously presented claims, the Examiner asserts that *Jaquette*, column 4, lines 42-50 teaches those features.

The combination of *O'Grady* and *Jaquette* does not teach or suggest these features of Applicants' amended claims. If, as the Examiner asserts, the CRC generating circuit 20 of *Jaquette* is analogous to Applicants' claimed CRC engine, then the CRC generating circuit 20 of *Jaquette* must be used to determine if an end one of the blocks is not full of data. *Jaquette* does not teach this.

Jaquette teaches "the last block of the compressed data is padded, if necessary, to make it exactly 32 bytes in length after the CRC is appended." *Jaquette*, column 4, lines 46-48. *Jaquette* does not teach CRC generator 20 determining if the last block of the compressed data needs to be padded.

Applicants also claim as said data is being written to each one of said plurality of blocks: calculating, by said CRC engine, a running cyclical redundancy code for each of said plurality of blocks by calculating an intermediate cyclical redundancy code after each byte of said data is written to each one of the plurality of blocks. After each intermediate cyclical redundancy code is calculated: the

intermediate cyclical redundancy code along with information that identifies said first port is stored, in a CRC memory that is included in the protocol interface device. The CRC memory is separate from but connected to said CRC engine. The combination of the references does not teach or suggest these features.

The Examiner states that *Fish* teaches implementing a write command for a data wherein the CRC is calculated during the writing and the calculated CRC is written at the end of the data. Thus, *Fish* teaches storing the calculated CRC at the end of the data.

In contradistinction, Applicants claim storing the intermediate cyclical redundancy code after each intermediate cyclical redundancy code is calculated. Each intermediate cyclical redundancy code is calculated after each byte of data is written to each one of the blocks. *Fish* does not teach storing intermediate cyclical redundancy codes. *Fish* does not teach storing intermediate cyclical redundancy codes after each intermediate cyclical redundancy code is calculated. And, *Fish* does not teach calculating an intermediate cyclical redundancy codes after each byte is written to each one of the blocks, and then storing intermediate cyclical redundancy codes after each intermediate cyclical redundancy code is calculated. Therefore, the combination of the cited references does not render Applicants' claims obvious.

In addition, Applicants claim storing the intermediate cyclical redundancy code along with information that identifies the first port. The combination of the cited references does not teach or suggest this feature.

It should also be noted that the intermediate cyclical redundancy code is stored in a CRC memory that is included in the protocol interface device, while the final value of the running cyclical redundancy code is stored in a second memory in the memory device. The memory device is connected to the protocol interface device. Therefore, the intermediate cyclical redundancy codes are stored in one device while the final value of the running cyclical redundancy codes is stored in a different device.

The remaining claims depend from one of the independent claims discussed above and are patentable for the reasons given above.

Applicants have amended claim 4 to recite: the protocol interface device including a selector switch that is coupled to said first port and said second port; coupling said selector switch to said CRC engine; using the selector switch to select one of the first port and the second port to which to couple the CRC engine, said CRC engine coupled to only one of the first and second ports at a time; and the CRC engine communicating with said first device and said second device only through said selector switch. The references do not teach a selector switch as claimed by Applicants whereby the CRC engine communicates with said first device and said second device only through said selector switch.

Applicants have added claim 15, which depends from claim 4, and recites: said selector switch being used to select the second port; in response to said second port being selected, receiving new data on the second port from said second device; said protocol interface device writing said new data, as said data is received, to successive ones of a second plurality of blocks until an end of said transfer length is reached, each one of said second plurality of blocks having a length of 2^n, where n is a positive integer; as said data is being written to each one of said second plurality of blocks: calculating, by said CRC engine, a second running cyclical redundancy code for each of said second plurality of blocks by calculating a second intermediate cyclical redundancy code after each byte of said data is written to each one of the plurality of blocks; after each second intermediate cyclical redundancy code is calculated: storing, in a CRC memory that is included in the protocol interface device, said second intermediate cyclical redundancy code along with information that identifies said second port. The combination of references does not teach or suggest these features.

Therefore, the rejection of claims 1, 4-5, 12 and 14 under 35 U.S.C. § 103(a) has been overcome.

I.B. Claims 2 and 13 over *O'Grady* in view of *Jaquette* and *Fish,* further in view of *Hogan*

The Examiner has rejected claims 2 and 13 as being unpatentable over *O'Grady* in view of *Jaquette* and *Fish* as applied to claims 1 and 12 above, and further in view of *Hogan* et al., Method and Apparatus for Discouraging Duplication of Digital Data, U.S. Patent No. 6,765,739, dated July 20, 2004 (hereinafter referred to as "*Hogan*"), This rejection, as it might be applied to the claims as amended, is respectfully traversed.

Claim 2 depends from claim 1. Claim 13 depends from claim 13. The combination of the cited references does not render Applicants' claims obvious for the reasons given above and because *Hogan* does not cure the deficiencies of the combination of *O'Grady*, *Jaquette*, and *Fish*. *Hogan* does not teach if the CRC engine determines that an end one of said plurality of blocks that includes an end of said transfer length is not full of data, adding padding; calculating an intermediate cyclical redundancy code after each byte is written to each one of the blocks; storing the intermediate cyclical redundancy code after each intermediate cyclical redundancy code is calculated; and storing the intermediate cyclical redundancy code along with information that identifies the first port.

Therefore, the rejection of claims 2 and 13 under 35 U.S.C. § 103(a) has been overcome.

I.C. Claim 3 over *O'Grady* in view of *Jaquette* and *Fish;* further in view of *PCTechGuide*

The Examiner has rejected claim 3 as being unpatentable over *O'Grady* in view of *Jaquette* and *Fish* as applied to claim above, and further in view of PCTechGuide - the PCTechnology Guide

(hereinafter referred to as "*PCTechGuide*"). This rejection, as it might be applied to the claim as amended, is respectfully traversed.

Claim 3 depends from claim 1. The combination of the cited references does not render Applicants' claim 3 obvious for the reasons given above and because *PCTechGuide* does not cure the deficiencies of the combination of *O'Grady*, *Jaquette*, and *Fish*. *PCTechGuide* does not teach if the CRC engine determines that an end one of said plurality of blocks that includes an end of said transfer length is not full of data, adding padding; calculating an intermediate cyclical redundancy codes after each byte is written to each one of the blocks; storing the intermediate cyclical redundancy code after each intermediate cyclical redundancy code is calculated; and storing the intermediate cyclical redundancy code along with information that identifies the first port.

Therefore, the rejection of claim 3 under 35 U.S.C. § 103(a) has been overcome.

I.D. Claims 7-8 and 10 over *O'Grady* in view of *Jaquette*, *Fish* and *Malakapalli*

The Examiner has rejected claims 7-8 and 10 over *O'Grady* in view of *Jaquette, Fish* and *Malakapalli* et al., Mass Storage Error Correction and Detection System, Method and Article of Manufacture, U.S. Patent No. 6,467,060, dated October 15, 2002 (hereinafter referred to as "*Malakapalli*"). This rejection, as it might be applied to the claims as amended, is respectfully traversed.

Applicants have amended claim 7 to describe if said CRC engine determines that an end one of said plurality of blocks that includes an end of said transfer length is not full of data, padding being added; said cyclical redundancy code engine calculating a running cyclical redundancy code by calculating an intermediate cyclical redundancy code after each byte of said data is written to each one of the plurality of blocks; and a CRC memory for, after each intermediate cyclical redundancy code is calculated, storing said intermediate cyclical redundancy code along with information that identifies one of the first or second ports through which said data was received.

Jaquette teaches "the last block of the compressed data is padded, if necessary, to make it exactly 32 bytes in length after the CRC is appended." *Jaquette*, column 4, lines 46-48. *Jaquette* does not teach padding the last block of the compressed data if CRC generator 20 determines that an end block is not full of data. Therefore, *Jaquette* does not teach if said CRC engine determines that an end one of said plurality of blocks that includes an end of said transfer length is not full of data, padding being added.

The Examiner states that *Fish* teaches implementing a write command for a data wherein the CRC is calculated during the writing and the calculated CRC is written at the end of the data. Thus, *Fish* teaches storing the calculated CRC at the end of the data.

In contradistinction, Applicants claim a CRC memory for, after each intermediate cyclical redundancy code is calculated, storing said intermediate cyclical redundancy code along with information

that identifies one of the first or second ports through which said data was received. The cyclical redundancy code engine calculates a running cyclical redundancy code by calculating an intermediate cyclical redundancy code after each byte of said data is written to each one of the plurality of blocks.

Fish does not teach storing intermediate cyclical redundancy codes. *Fish* does not teach storing intermediate cyclical redundancy codes after each intermediate cyclical redundancy code is calculated. And, *Fish* does not teach calculating an intermediate cyclical redundancy code after each byte is written to each one of the blocks, and then storing the intermediate cyclical redundancy code after each intermediate cyclical redundancy code is calculated. Therefore, the combination of the cited references does not render Applicants' claims obvious.

In addition, Applicants claim storing the intermediate cyclical redundancy code along with information that identifies the first port. The combination of the cited references does not teach or suggest this feature.

The remaining claims depend from claim 7 and are patentable for the reasons given above.

Therefore, the rejection of claims 7-8 and 10 under 35 U.S.C. § 103(a) has been overcome.

I.E. Claim 9 over *O'Grady* in view of *Jaquette, Fish* and *Malakapalli* as applied to claim 7 above, and further in view of *Hogan*

The Examiner has rejected claim 9 over *O'Grady* in view of *Jaquette, Fish* and *Malakapalli* as applied to claim 7 above, and further in view of *Hogan.* This rejection, as it might be applied to the claims as amended, is respectfully traversed.

Claim 9 depends from claim 7. The Examiner relies on *Malakapalli* to cure the deficiencies of the combination of *O'Grady*, *Jaquette*, and *Fish*. *Malakapalli* does not cure the deficiencies because *Malakapalli* does not teach if said CRC engine determines that an end one of said plurality of blocks that includes an end of said transfer length is not full of data, padding being added; said cyclical redundancy code engine calculating a running cyclical redundancy code by calculating an intermediate cyclical redundancy code after each byte of said data is written to each one of the plurality of blocks; and a CRC memory for, after each intermediate cyclical redundancy code is calculated, storing said intermediate cyclical redundancy code along with information that identifies one of the first or second ports through which said data was received. Therefore, the rejection of claim 9 under 35 U.S.C. § 103(a) has been overcome.

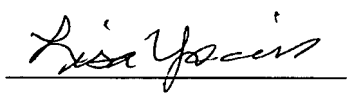
II. Conclusion

It is respectfully urged that the subject application is patentable over the cited prior art and is now in condition for allowance.

The examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: August 14, 2007

Respectfully submitted,



Lisa L.B. Yociss
Reg. No. 36,975
Yee & Associates, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 385-8777
Attorney for Applicant